Amendments to the Claims:

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1   1.   (Currently amended): A data transformation system comprising:

2   a computing device;

3   a data interface on the computing device configured to receive data to be

4 transformed or to send transformed data;

5   memory configured to store one or more transform process definitions having at

6 least one simple transform definition and at least one compound transform definition;

7   an application including computer instructions; and

8   a data interpreter configured to exchange data with the data interface and the

9 application, the data interpreter including a transform engine configured to

10   select a first compound transform ~~process~~ definition from the one or more

11 transform process definitions, the selected first compound transform ~~process~~ definition including

12 a hierarchical data structure with a first sub-definition,

13   invoke a first parallel processing thread to process the first compound transform

14 definition including the first sub-definition,

15   select a second compound transform definition from the one or more transform

16 process definitions, the selected second compound transform definition including a hierarchical

17 data structure with a second sub-definition,

18   invoke a second parallel processing thread to process the second compound

19 transform definition including the second sub-definition,

20   concurrently navigate the selected first compound transform ~~process~~ definition,

21 the selected second compound transform definition, and the data to be transformed using the

22 parallel processing threads, navigation within the data to be transformed being responsive to

Appl. No. 10/661,167
Amdt. dated May 27, 2009
Reply to Office Action of April 1, 2009

PATENT

23 | transform definitions within the selected <u>first and second</u> transform <u>definitions</u> ~~process~~

24 | ~~definition~~, and

25   generate output data having a data structure responsive to a data structure of the

26 | selected <u>first and second compound</u> transform <u>definitions</u> ~~process definition~~.

1   2.   (Original): The data transformation system of claim 1, wherein the

2 transformation engine is further configured to process the at least one compound transform

3 definition using recursion.

1   3.   (Original): The data transformation system of claim 1, wherein the data

2 interpreter is further configured to support a plurality of applications.

1   4.   (Original): The data transformation system of claim 1, wherein the

2 application is a database application, accounting application, human resources application,

3 customer management application, inventory application, or an internet application.

1   5.   (Original): The data transformation system of claim 1, wherein the

2 application and the data interpreter are integrated.

1   6.   (Original): The data transformation system of claim 1, wherein the data

2 interpreter further includes a computing device configured to support the transform engine.

1   7.   (Currently amended): A data interpreter configured to transform data to

2 be transformed, the data interpreter comprising:

3   at least one computing device; and

4   a transform engine supported by the computing device, the transform engine

5 being configured to

6   access a transform process definition including a hierarchical data structure of

7 | transform definitions, the data structure including a simple transform definition<u>,</u> ~~and~~ a <u>first</u>

8 | compound transform definition <u>having a first sub-definition, and a second compound transform</u>

9 | <u>definition having a second sub-definition,</u>

10   invoke a first parallel processing thread to process the first compound transform

11 definition including the first sub-definition,

12   invoke a second parallel processing thread to process the second compound

13 transform definition including the second sub-definition,

14   concurrently navigate the ~~transform process definition~~ first compound transform

15 definition, the second compound transform definition, and the data to be transformed using the

16 parallel processing threads, navigation within the data to be transformed being responsive to the

17 first and second compound transform definitions ~~definition~~ within the transform process

18 definition, and

19   generate output data having a data structure responsive to the first and second

20 compound transform definitions ~~process definition~~.

1   8.   (Original): The data interpreter of claim 7, wherein the data structure of

2 the output data is responsive to a structure of transform process definition.

1   9.   (Original): The data interpreter of claim 7, wherein the transform engine

2 is configured to process the compound transform definition using recursion.

1   10.   (Original): The data interpreter of claim 7, wherein the transform engine

2 is configured to generate output data including data elements characterized by the transform

3 process definition and having no contribution from the data to be transformed.

1   11.   (Original): The data interpreter of claim 7, wherein the transform process

2 definition is configured such that some data elements in the data to be transformed do not make a

3 contribution to the output data.

1   12.   (Currently amended): The data interpreter of claim 7, wherein the

2 transform engine is further configured to navigate the data to be transformed responsive to the

3 ~~date~~ data structure of the transform definitions within the transform process definition.

1   13.   (Original): The data interpreter of claim 7, wherein the transform engine

2   is further configured to navigate the data to be transformed responsive to content of the

3   transform definitions.

1   14.   (Currently amended): A method of transforming data using an application

2   programming interface, the method comprising:

3   receiving data to be transformed at the application programming interface;

4   parsing identification data within the data to be transformed, the identification

5   data characterizing the data to be transformed;

6   using the identification data to select a first compound transform process

7   definition from a set of one or more transform process definitions, the selected first compound

8   transform process definition having a first sub-definition and defining a process of translating

9   data elements within the data to be transformed to output data elements; and

10   invoking a first parallel processing thread to process the first compound transform

11 definition including the first sub-definition;

12   using the identification data to select a second compound transform definition

13 from the set of one or more transform process definitions, the selected second compound

14 transform definition having a second sub-definition and defining a process of translating data

15 elements within the data to be transformed to output data elements;

16   invoking a second parallel processing thread to process the second compound

17 transform definition including the second sub-definition; and

18   transforming the data to be transformed to output data, using a transformation

19 engine and the selected first and second compound transform definitions process definition, a

20 data structure of the output data being responsive to a data structure of the first and second

21 compound transform definitions process definition.

1   15.   (Original): The method of claim 14, wherein transforming the data to be

2   transformed includes nesting of data records.

1   16.   (Currently amended): The method of claim 14, wherein transforming the

2 data to be transformed includes calling a transformation process recursively responsive to a

3 compound transform definition in the selected <u>first</u> transform ~~process~~ definition.

1   17.   (Currently amended): The method of claim 14, wherein the selected <u>first</u>

2 <u>and second</u> transform <u>definitions</u> ~~process definition is~~ <u>are</u> selected based on information within

3 the identification data that identifies a destination of the data to be transformed.

1   18.   (Currently amended): The method of claim 14, wherein the <u>first and</u>

2 <u>second</u> transform <u>definitions</u> ~~process definition is~~ <u>are</u> selected based on information within the

3 identification data that identifies a format of the output data.

1   19.   (Original): The method of claim 14, wherein the transform process

2 definition includes a extensible markup language (XML).

1   20.   (Currently amended): A method of transforming data using an application

2 programming interface, the method comprising:

3   receiving data to be transformed at the application programming interface, the

4 data to be transformed including identification data;

5   using the identification data to select a <u>first compound</u> transform ~~process~~

6 definition from a set of transform process definitions, the selected <u>first compound</u> transform

7 ~~process~~ definition <u>having a first sub-definition and</u> defining a process of translating data

8 elements within data to be transformed to output data elements; ~~and~~

9   <u>invoking a first parallel processing thread to process the first compound transform</u>

10 <u>definition including the first sub-definition;</u>

11   <u>using the identification data to select a second compound transform definition</u>

12 <u>from the set of one or more transform process definitions, the selected second compound</u>

13 <u>transform definition having a second sub-definition and defining a process of translating data</u>

14 <u>elements within the data to be transformed to output data elements;</u>

15   invoking a second parallel processing thread to process the second compound

16   transform definition including the second sub-definition; and

17   transforming the data to be transformed to output data by concurrently navigating

18   the data to be transformed, the first compound transform definition, and the ~~selected~~ second

19   compound transform definition using the parallel processing threads ~~process definition~~,

20   navigation in the data to be transformed being responsive to the transform process definition.

1   21.   (Currently amended): The method of claim 20, wherein the selected first

2   and second compound transform definitions ~~process definition is~~ are selected based on

3   information within the identification data that identifies a format of the data to be transformed.

1   22.   (Currently amended): The method of claim 20, wherein the selected first

2   and second compound transform definitions ~~process definition is~~ are selected based on

3   information within the identification data that identifies a source of the data to be transformed.

1   23.   (Original): The method of claim 20, wherein the application programming

2   interface is shared by several applications.

1   24.   (Currently amended): The method of claim 20, wherein the application

2   programming interface is shared by several applications and the selected first and second

3   compound transform definitions ~~process definition is~~ are selected based on an identity of one of

4   the several applications.

1   25.   (Currently amended): The method of claim 20, wherein a structure of the

2   output data is responsive to a structure of a transform definition included in the selected first and

3   second compound transform definitions ~~process definition~~.

1   26.   (Original): The method of claim 20, wherein transforming the data to be

2   transformed includes nesting of data records.

1   27.   (Original): The method of claim 20, wherein transforming the data to be

2   transformed includes filtering of data records.

1   28.   (Currently amended): The method of claim 20, wherein transforming the

2   data to be transformed includes calling a transformation process recursively responsive to a data

3   structure of a transform definition included in the selected <u>first and second compound</u> transform

4   definitions~~ process definition~~.

1   29.   (Original): The method of claim 20, wherein transforming the data to be

2   transformed includes searching the data to be transformed for a data field specified in a

3   transform definition included in the transform process definition.

1   30.   (Original): The method of claim 20, wherein the transform definition

2   includes a translation codeset parameter configured to invoke an external reference.

1   31.   (Original): The method of claim 20, wherein the transform definition

2   includes a translation codeset parameter configured to invoke an external reference, the external

3   reference being configured to perform logic operations using the data to be transformed.

1   32.   (Currently amended): A method of transforming data, the method

2   comprising:

3   positioning a definition pointer to point at ~~one~~ <u>a first compound transform</u>

4   <u>definition</u> ~~of a plurality of transform definitions~~ within a transform process definition;

5   <u>invoking a first parallel processing thread to read</u> ~~reading~~ the pointed at <u>first</u>

6   <u>compound</u> transform definition;

7   searching data to be transformed for a data element to be transformed, the search

8   being responsive to the ~~pointed at~~ <u>first compound</u> transform definition; ~~and~~

9   transforming any found data element into output data, responsive to the ~~pointed at~~

10 <u>first compound</u> transform definition, a data structure of the output data being responsive to a data

11 structure of the transform process definition;

12   <u>positioning a definition pointer to point at a second compound transform</u>

13 <u>definition within the transform process definition;</u>

14   invoking a second parallel processing thread to read the pointed at second

15   compound transform definition;

16   searching data to be transformed for another data element to be transformed, the

17   search being responsive to the second compound transform definition; and

18   transforming any found data element into output data, responsive to the second

19   compound transform definition, the data structure of the output data being responsive to the data

20   structure of the transform process definition,

21   wherein the read first compound transform definition includes a translation

22   codeset parameter enabling the transforming to include a call to one of a function or a lookup

23   table located in the ~~pointed at~~ first compound transform definition.

1   33.   (Currently amended): The method of claim 32, further including

2   determining a type of the read first compound transform definition and, if the first compound

3   transform definition is not a simple transform definition type, recursively calling the method of

4   claim 32.

1   34.   (Original): The method of claim 32, further including determining if all

2   sub-definitions of a compound transform definition have been processed.

1   35.   (Original): The method of claim 32, wherein the method of transforming

2   data includes nesting of a data element.

1   36.   (Currently amended): The method of claim 32, further including, if no

2   data element is found in ~~the~~ either step of searching data to be transformed, adding an output data

3   element to the output data responsive to the read first compound transform definition, the data to

4   be transformed having no contribution to the output data element.

1   37.   (Currently amended): The method of claim 32, wherein the read first

2   compound transform definition includes a value parameter configured to specify a value for

3   inclusion in the output data.

Appl. No. 10/661,167
Amdt. dated May 27, 2009
Reply to Office Action of April 1, 2009

PATENT

1   38.   (Original): The method of claim 32, wherein the data element is a

2   compound data element and the read transform definition includes a source record parameter

3   configured to specify the compound data element.

1   39.   (Currently amended): The method of claim 32, wherein the read <u>first</u>

2   <u>compound</u> transform definition is in a meta-language format.

1   40.   (Original): The method of claim 32, wherein the data to be transformed

2   data is in a meta-language data format.

1   41.   (Currently amended): The method of claim 32, wherein the read <u>first</u>

2   <u>compound</u> transform definition includes a transform element having an output field name and a

3   source field parameter.   .

1   42.   (Currently amended): The method of claim 32, wherein the read <u>first</u>

2   <u>compound</u> transform definition includes a value parameter configured to populate a field in the

3   output data.

1   43.   (Canceled)

1   44.   (Currently amended): A method of transforming data, the method

2   comprising:

3   positioning a definition pointer to point at a <u>first compound</u> transform definition,

4   the <u>first compound</u> transform definition being ~~one of a plurality of transform definitions~~ within a

5   transform process definition;

6   <u>invoking a first parallel processing thread to read</u> ~~reading~~ the pointed at <u>first</u>

7   <u>compound</u> transform definition <u>and sub-definitions of the first compound transform definition</u>;

8   positioning a <u>first</u> payload pointer to point at a data element to be transformed, the

9   positioning being responsive to a data structure of the <u>first compound</u> transform ~~process~~

10   definition;

Appl. No. 10/661,167
Amdt. dated May 27, 2009
Reply to Office Action of April 1, 2009

PATENT

11   transforming the data element into output data, responsive to the read first

12   compound transform definition; ~~and~~

13   positioning the definition pointer to point at a second compound transform

14   definition, the second compound transform definition being within the transform process

15   definition;

16   invoking a second parallel processing thread to read the pointed at second

17   compound transform definition and sub-definitions of the second compound transform

18   definition;

19   positioning a second payload pointer to point at a data element to be transformed,

20   the positioning being responsive to a data structure of the second compound transform definition;

21   and

22   transforming the second data element into output data, responsive to the read

23   second compound transform definition.

24   ~~invoking a parallel process to process sub-definitions of the read transform~~

25   ~~definition.~~

1   45.   (Currently amended): The method of claim 44, further including

2   determining a type of the read first compound transform definition and, if the read first

3   compound transform definition is not a simple transform definition type, recursively calling the

4   method of claim 44.

1   46.   (Currently amended): The method of claim 44, further including

2   determining a type of the read first compound transform definition and, if the read first

3   compound transform definition is not a simple transform definition type, recursively calling the

4   method of claim 44, wherein the recursive call is responsive to the data structure of the transform

5   process definition.

1   47.   (Currently amended): The method of claim 44, further including

2   determining a type of the read first compound transform definition, if the read transform

Appl. No. 10/661,167
Amdt. dated May 27, 2009
Reply to Office Action of April 1, 2009

PATENT

3   definition is not a simple transform definition type recursively calling the method of claim 44,

4   and determining if all sub-elements of a compound element have been transformed.

1   48.   (Original): The method of claim 44, further including determining if all

2   sub-elements of a compound element have been transformed and, if the determination returns a

3   value of YES, returning to a calling process.

1   49.   (Canceled)

1   50.   (Original): The method of claim 44, wherein the method of transforming

2   data includes un-nesting of the data element to be transformed.

1   51.   (Currently amended): The method of claim 44, wherein the read <u>first</u>

2   <u>compound</u> transform definition includes a source field parameter configured to specify the data

3   element.

1   52.   (Currently amended): The method of claim 44, wherein the read <u>first</u>

2   <u>compound</u> transform definition includes a source record parameter configured to specify the

3   compound data element.

1   53.   (Currently amended): The method of claim 44, wherein the read <u>first</u>

2   <u>compound</u> transform definition includes a translation codeset configured for calling computer

3   instructions including logic operations.

1   54.   (Original): The method of claim 53, wherein the computer instructions are

2   configured to call an external process.

1   55.   (Original): The method of claim 44, further including a step of combining

2   the data element with the transform process definition prior to transforming the data element to

3   output data.

1   56.   (Original): The method of claim 44, wherein the transform process

2   definition includes a tree data structure.

1   57.   (Currently amended): A method of preparing data for transformation, the

2   method comprising:

3   receiving data to be transformed;

4   parsing the received data to determine identification information;

5   invoking a parallel processing thread to use using the identification information to

6   extract a first compound transform process definition from a plurality of transform process

7   definitions, the extracted first compound transform process definition including a transform

8   definition configured to transform the data to be transformed, to direct navigation within the data

9   to be transformed during transformation, and to determine a data structure of output data

10   resulting from transformation of the data to be transformed, the transform definition including a

11   hierarchical data structure having at least one simple transform definition and at least one

12   compound transform definition, the first compound transform definition being configured to

13   generate a compound data element in the output data; and

14   invoking a parallel processing thread to use the identification information to

15   extract a second compound transform definition from the plurality of transform process

16   definitions, the extracted second compound transform definition including a transform definition

17   configured to transform the data to be transformed, to direct navigation within the data to be

18   transformed during transformation, and to determine the data structure of output data resulting

19   from transformation of the data to be transformed, the second compound transform definition

20   being configured to generate a compound data element in the output data; and

21   adding the extracted transform process definition to meta-language transform

22   input data including the data to be transformed.

1   58.   (Original): The method of claim 57, wherein the extracted transform

2   process definition is in a meta-language format.

1   59.   (Currently amended): A computer readable storage media having

2   embodied thereon data, the data comprising:

3   computer instructions configured to position a definition pointer to point at a <u>first</u>

4   <u>compound</u> transform definition, the <u>first compound</u> transform definition being ~~one of a plurality~~

5   ~~of transform definitions~~ within a transform process definition;

6   computer instructions configured to <u>invoke a first parallel processing thread to</u>

7   read the pointed at <u>first compound</u> transform definition <u>and sub-definitions of the first compound</u>

8   <u>transform definition</u>;

9   computer instructions configured to increment a <u>first</u> payload pointer, within the

10 data to be transformed, to a data element to be transformed, the incrementation being responsive

11 to the pointed at <u>first compound</u> transform definition;

12 computer instructions configured to transform any found data element into output

13 data, responsive to the ~~pointed at~~ <u>first compound</u> transform definition<u>;</u> ~~a data structure of the~~

14 ~~output data being responsive to a data structure of the transform process definition; and~~

15 <u>computer instructions configured to increment a second payload pointer, within</u>

16 <u>the data to be transformed, to a data element to be transformed, the incrementation being</u>

17 <u>responsive to the pointed at second compound transform definition;</u>

18 <u>computer instructions configured to invoke a second parallel processing thread to</u>

19 <u>read the pointed at second compound transform definition and sub-definitions of the second</u>

20 <u>compound transform definition; and</u>

21 <u>computer instructions configured to transform the second data element into output</u>

22 <u>data, responsive to the read second compound transform definition.</u>

23 ~~computer instructions configured to invoke parallel processes to position the~~

24 ~~definition pointer and increment the payload pointer such that positioning the definition pointer~~

25 ~~and incrementing the payload pointer are enabled to occur concurrently.~~

1   60.   (Original): The computer readable media of claim 59, wherein the data

2   further comprises computer instructions configured to employ recursion to transform a

3   compound data element within the data to be transformed.

Appl. No. 10/661,167
Amdt. dated May 27, 2009
Reply to Office Action of April 1, 2009

PATENT

1   61.   (Original): The computer readable media of claim 59, wherein the data

2   further comprises computer instructions configured to transform the data to be transformed using

3   parallel processes.

1   62.   (Currently amended): A computer readable storage media having

2   embodied thereon data, the data comprising:

3   payload data including data to be transformed, the data to be transformed

4   including metadata characterizing simple data elements and compound data elements; and

5   a transform process definition including a <u>first and a second compound</u> transform

6   definition configured to transform the data to be transformed, to direct navigation within the data

7   to be transformed during transformation, and to determine a data structure of output data

8   resulting from the transformation, ~~the transform definition including a hierarchical data structure~~

9   ~~having at least one simple transform definition and at least one compound transform definition,~~

10   the <u>first compound</u> transform definition also including a pointer to a function enabled to perform

11   logical operations on the data to be transformed during transformation and generate the output

12   data, the <u>first</u> compound transform definition being configured to generate a compound data

13   element in the output data <u>using parallel processing threads configured to navigate the first and</u>

14   <u>second compound transform definitions concurrently</u>.

1   63.   (Original): The computer readable media of claim 62, wherein the

2   computer readable media includes memory included in a data interface.

1   64.   (Original): The computer readable media of claim 62, wherein the

2   computer readable media includes a hard drive.

1   65.   (Currently amended): An application system comprising:

2   a computing device;

3   means for positioning a definition pointer to point at a <u>first compound</u> transform

4   definition within a transform process definition;

5   ·   means for <u>invoking a first parallel processing thread to read</u> ~~reading~~ the <u>first</u>

6 <u>compound</u> transform definition by the computing device;

7   <u>means for positioning the definition pointer to point at a second compound</u>

8 <u>transform definition within the transform process definition;</u>

9   <u>means for invoking a second parallel processing thread to read the second</u>

10 <u>compound transform definition by the computing device;</u>

11   means for positioning a payload pointer to point to a first data element, the first

12 data element being a member of a plurality of data elements within data to be transformed; and

13   means for generating output data using the first data element and the <u>first and</u>

14 <u>second compound</u> transform ~~definition~~ <u>definitions</u>;

15   wherein the means for positioning the definition pointer and the means for

16 positioning the payload pointer are enabled to be invoked concurrently.

1   66.   (Original): The application system of claim 65, further including means

2 for selecting the transform process definition from a set of transform process definitions,

3 responsive to data associated with the data to be transformed.

1   67.   (Original): The application system of claim 65, wherein a second data

2 element has no contribution to output data generated using the transform process definition, the

3 second data element being a member of the plurality of data elements.

1   68.   (Original): The application system of claim 65, further including means

2 for adding data to the output data, the added data being configured responsive to the transform

3 process definition and having no contribution from the data to be transformed.